

## REMARKS

In the November 22, 2004, Office Action (hereinafter "Office Action"), Claims 1-31 were rejected under 35 U.S.C. § 102(e) as being anticipated by U.S. Patent No. 6,721,941 to Morshed et al. (hereinafter "Morshed"). With this response, Claims 1-31 remain pending. Claims 32-34 are newly added.

Pursuant to 37 C.F.R. § 1.111, and for the reasons set forth below, applicants respectfully request reconsideration and allowance of this application.

Prior to discussing why applicants believe that the pending claims are allowable, a brief description of the present invention and the cited reference is presented. The discussion of the present invention and the cited reference is not provided to define the scope or interpretation of any of the claims of this application. Rather, the discussion is provided to help the U.S. Patent and Trademark Office better appreciate important claim distinctions between the present invention and the cited reference.

### Background and Summary of the Invention

As a background, writing software modules to run in a managed code environment provides many benefits to a software developer. Some of the benefits includes portability (i.e., the software module runs on any machine or system to which the managed code environment is ported), a rich selection of services, easy development cycles, and the like. Unfortunately, managed code applications exist only within the managed code environment. This is not a problem unless an application or service outside of the managed code environment wants to access a managed code application. For example, a system administrator, operating on a remote computer, may need to monitor values generated by a managed code application operating on a user computer. Unfortunately, such access currently only reaches to the managed code environment, not the managed code applications executing in that environment.

LAW OFFICES OF  
CHRISTENSEN O'CONNOR JOHNSON KINDNESS<sup>PLLC</sup>  
1420 Fifth Avenue  
Suite 2800  
Seattle, Washington 98101  
206.682.8100

In this light, the present invention is directed at solving the problem of remotely accessing a software object executing within a managed code environment, and more particularly, to accessing instrumentation data of a managed code application by an instrumentation data source executing outside of the managed code environment. It should be noted that the external application is referred to as an instrumentation data source, as this application (the information data source) is typically the focal point, or source, for yet other applications to obtain information regarding the managed code application. As described in the application, in one embodiment, instrumentation data source is implemented as a Windows Management Instrumentation module that is used to provide enterprise-based management.

According to the present invention, when there is an indication that a managed code application has been configured such that it can provide instrumentation data (such as data used for remote monitoring and management of the application) to an external information data source, a decoupled provider is loaded within the managed code environment. The decoupled provider facilitates/enables communication between the managed code application and the external instrumentation data source. In other words, the decoupled provider, residing within the managed code environment, is configured in such a way as to be able to communicate with other objects external to the managed code environment.

As those skilled in the art will appreciate, each managed code object exposes different instrumentation data according to implementation details. As each is different, a schema describing the particulars of the managed code application must be registered with the instrumentation data source. Thus, with the decoupled provider loaded, the decoupled provider determines whether the schema describing the instrumentation data of the managed code application is registered with the instrumentation data source. Furthermore, if the decoupled provider determines that the schema has been registered with the data source, the decoupled provider determines whether the schema registered with the instrumentation data source is

LAW OFFICES OF  
CHRISTENSEN O'CONNOR JOHNSON KINDNESS<sup>PLLC</sup>  
1420 Fifth Avenue  
Suite 2800  
Seattle, Washington 98101  
206.682.8100

up-to-date, i.e., is correct/current, and if not, registers the latest schema with the instrumentation data source.

Once the decoupled provider is loaded in the managed code environment, and the managed code application's schema is registered with the instrumentation data source, instrumentation data can be retrieved from and set at the managed code application through the "conduit" provided by the decoupled provider.

#### Morshed (U.S. Patent No. 6,721,941)

As those skilled in the art will appreciate, with regard to debugging a software application, "instrumented code" refers to code that has been modified to generate debugging information as the modified application is executed. Debugging information may include data store values, timing information, counters, memory usage, system service accesses, and the like, all determined according to how the developer modified/instrumented the code. This debugging information is typically analyzed and used to remove bugs from the software module and to improve efficiency of that software module.

Morshed describes a debugging system, one that "instruments" software modules such that the software modules, when executed, generate debug information such as that described above. Moreover, Morshed describes the details of "instrumenting" a code module such that debugging information is generated.

One aspect described in the Morshed system is a database storing data regarding objects and their interconnections as gathered by a monitoring process. (See Morshed, Col. 73, lines 1-15.) As this information is stored in a database, Morshed further discloses a database schema that describes how the database is organized. (See Morshed, Col. 73, lines 15-21.) In other words, unlike the present invention, this schema described the organization of the database, not a schema for obtaining/understanding the instrumentation data of a managed code object.

While the Morshed system purportedly operates on both native code and managed code applications, Morshed is directed at instrumenting a code module with debugging information such that, when executed, debugging information is made available and utilized. Purportedly, the debugging tool is able to obtain debugging information as instrumented code modules execute. However, even if the Morshed system operates on managed code applications, Morshed fails to address the issue of accessing information regarding the managed code application by an external application concurrently executing outside of the managed code environment. As those skilled in the art will appreciate, the debugging tools either (a) operate within the managed code environment, or (b) represent the managed code environment. Morshed simply fails to disclose accessing the instrumentation data of a managed code application executing in a managed code environment by an external application concurrently executing outside of that environment.

### The Claims Distinguished

#### Claim 1

The Office Action rejected Claim 1 as being anticipated by Morshed. Applicants respectfully disagree, and assert that Morshed fails to disclose each element of Claim 1. In particular, applicants assert that Morshed fails to disclose the following elements of Claim 1:

"loading a decoupled provider for facilitating communication between said managed code environment and said instrumentation data source,"

"determining whether schema describing said instrumentation data has been previously registered with said instrumentation data source," and

"in response to determining that said schema has not been previously registered with said instrumentation data source, registering said schema with said instrumentation data source through said decoupled provider."

In regard to the element "loading a decoupled provider for facilitating communication between said managed code environment and said instrumentation data source," the Office

LAW OFFICES OF  
CHRISTENSEN O'CONNOR JOHNSON KINDNESS<sup>PLLC</sup>  
1420 Fifth Avenue  
Suite 2800  
Seattle, Washington 98101  
206.682.8100

Action cites to Morshed, Col. 16, lines 40-47. Applicants assert that the Office Action has misconstrued the cited passage. More particularly, Morshed, Col. 16, lines 40-47, is specifically directed to a portion of a process for determining which objects within a given source module are to be instrumented, i.e., which objects are to be associated with debugging information. This citation in Morshed completely fails to disclose "**loading** a decoupled provider," and moreover, fails to disclose loading a decoupled provider that facilitates communication between a managed code application and an instrumentation data source. Simply stated, this particular passage describes a branch in a method based on a determination as to whether data in a compiled source object should be associated with debugging information.

In regard to the element "determining whether schema describing said instrumentation data has been previously registered with said instrumentation data source," the Office Action cites to Morshed, Col. 73, lines 15-25. Applicants acknowledge that this passage in Morshed references a "schema." However, the "schema" referenced by Morshed describes the organization of a database into which objects are stored. The schema referenced by Morshed does not describe instrumentation data of a managed code application executing within a managed code environment. Furthermore, Morshed, Col. 73, lines 15-25, utterly fails to describe determining whether the schema is registered with the data source. There is absolutely no description of a determination as to whether the schema is registered.

In regard to the element "determining that said schema has not been previously registered with said instrumentation data source, and registering said schema with said instrumentation data source through said decoupled provider," the Office Action cites to Morshed, Col. 54, lines 1-15. In the first place, even if Morshed disclosed a schema of a managed code application's instrumentation data, which applicants expressly deny, this cited passage does not reference the same schema that the Office Action previously identified as the schema. Indeed, Morshed fails

LAW OFFICES OF  
CHRISTENSEN O'CONNOR JOHNSON KINDNESS<sup>PLLC</sup>  
1420 Fifth Avenue  
Suite 2800  
Seattle, Washington 98101  
206.682.8100

to disclose determining whether the database schema of Col. 73, lines 15-25, is registered with the instrumentation data source.

Morshed, Col. 54, lines 1-15, relates to monitoring remote procedure calls (RPCs), and the various objects used in correlating information between the processes of the RPCs. Morshed fails to disclose determining whether a schema has been previously registered with an instrumentation data source. Moreover, Morshed fails to disclose registering the schema (of a managed code application) with an instrumentation data source through a decoupled provider.

As discussed above, applicants assert that Morshed fails to disclose each element of independent Claim 1. A claim is anticipated only when each and every element of the Claim is disclosed in a single prior art reference. See, *Verdegaal Bros. v. Union Oil Co. Of California*, 814 F.2d, 628, 631, 2 U.S.P.Q.2d 1051, 1053 (Fed. Cir. 1987). Accordingly, applicants submit that the 35 U.S.C. § 102(e) rejection of Claim 1 is in error, and request that the rejection be withdrawn and the claim allowed.

#### Claim 2

Claim 2 depends from independent Claim 1. Accordingly, for the same reasons described above in regard to Claim 1, applicants assert that Claim 2 is allowable over Morshed, especially when read in combination with Claim 1.

Additionally, applicants assert that Claim 2 includes elements that are not disclosed in Morshed. In particular, applicants assert that Morshed fails to disclose the following elements:

"in response to determining that said schema has previously been registered with said instrumentation data source, determining whether said previously registered schema is correct," and

"in response to determining that said previously registered schema is incorrect, overwriting said previously registered schema with said schema describing said instrumentation data."

In regard to the element "in response to determining that said schema has previously been registered with said instrumentation data source, determining whether said previously registered schema is correct," the Office Action cites to Morshed, Col. 34, lines 46-50. This passage in Morshed relates to obtaining profiling information regarding an executing application. As those skilled in the art will readily appreciate, profiling, as used by Morshed, refers to generating information regarding the actual execution of an application. Clearly, this cited passage is not related to determining whether a schema has been previously registered with an information data source. Furthermore, there is no disclosure in Morshed of "determining whether said previously registered schema is correct."

In regard to the element "in response to determining that said previously registered schema is incorrect, overwriting said previously registered schema with said schema describing said instrumentation data," the Office Action cites to Morshed, Col. 28, lines 37-44, and, in particular, mentions updating invalid instructions. This particular section of Morshed discusses how various tables must be updated due to the addition of instrumentation data, i.e., debugging information, into the source code. In particular, Morshed explicitly states, on line 44, "since instrumentation modifies the byte code offsets, then the line number table requires modification at the step 620 to reflect the offset change." Updating byte offsets due to the addition of instrumentation data is not the same as registering a schema with an application data source. Clearly, this passage can hardly be viewed as "determining that said previously registered schema is incorrect, overwriting said previously registered schema with said schema describing said instrumentation data," as described in Claim 2.

For the reasons described above, applicants respectfully assert that Morshed fails to disclose each and every element of Claim 2. Accordingly, applicants request that the 35 U.S.C. § 102(e) rejection of Claim 2 be withdrawn and the claim allowed.

LAW OFFICES OF  
CHRISTENSEN O'CONNOR JOHNSON KINDNESS<sup>PLLC</sup>  
1420 Fifth Avenue  
Suite 2800  
Seattle, Washington 98101  
206.682.8100

### Claim 5

Claim 5 depends from independent Claim 1. Accordingly, for the same reasons described above in regard to Claim 1, applicants assert that Claim 5 is allowable over Morshed, especially when read in combination with Claim 1.

In addition to depending from independent Claim 1, applicants assert that Morshed fails to disclose each element of this claim. More particularly, applicants assert that Morshed fails to disclose the element "**said instrumentation data comprises an object** and wherein said schema describes the **properties and methods exposed by said object**."

The Office Action cites to Morshed, Col. 73, lines 18-25. As already discussed, this passage of Morshed generally describes that a schema describes how information in a database is stored. This passage of Morshed utterly fails to disclose that the instrumentation data is an object, and that the schema describing that instrumentation data describes the properties and methods exposed by that object. Accordingly, applicants submit that the 35 U.S.C. § 102(e) rejection of Claim 5 is error, and request that the rejection be withdrawn and the claim allowed.

### Claims 3, 4, 6, and 7

Claims 3, 4, 6, and 7 each depend from independent Claim 1. Accordingly, for the same reasons described above in regard to Claim 1, applicants assert that these claims are allowable over Morshed, especially when read in combination with Claim 1. Applicants request the 35 U.S.C. § 102(e) rejection of Claims 3, 4, 6, and 7 be withdrawn, and the claims allowed.

### Claims 8 and 9

Claims 8 and 9 were rejected for the same reasons as Claims 1-7. Accordingly, for the same reasons as described above, applicants assert that the 35 U.S.C. § 102(e) rejection of Claims 8 and 9 was in error, and request that the rejections be withdrawn, and the claims allowed.

### Claim 10

The Office Action rejected Claim 10 as being anticipated by Morshed. Applicants respectfully disagree, and assert that Morshed fails to disclose each element of Claim 10. In particular, applicants assert that Morshed fails to disclose the following elements of Claim 10:

"loading a decoupled provider for facilitating communication between said managed code environment and said instrumentation data source,"

"determining whether schema describing said instrumentation data has been previously registered with said instrumentation data source,"

"in response to determining that said schema has not been previously registered with said instrumentation data source, registering said schema with said instrumentation data source through said decoupled provider,"

"receiving a request for said instrumentation data from said instrumentation data source at said decoupled proxy," and

"converting said instrumentation data from a format compatible with said managed code environment to a format compatible with said instrumentation data source, and transmitting said converted instrumentation data to said instrumentation data source."

Applicants note that the first three elements recited above are the same as those discussed in regard to Claim 1. Thus, for the same reasons described above, applicants assert that Claim 10 is allowable over Morshed.

In regard to the element "receiving a request for said instrumentation data from said instrumentation data source at said decoupled proxy," the Office Action cites to Morshed, Col. 57, lines 52-62. This passage does not relate to the method previously referenced by the Office Action as disclosing a decoupled provider (i.e., Morshed, Col. 16, lines 40-47.) In this cited passage, a monitor process is established for "relaying method calls ... between a local monitor process and the remote collector." Morshed, Col. 57, lines 58-60. This is very confusing. On the one hand, the Office Action first asserts that the decoupled proxy is a method

for determining which tuples are to be associated with debugging information. Then, the Office Action asserts that the decoupled proxy is a relay mechanism for relaying messages between two processes. Applicants assert that the Office Action has cited to nothing more than a message relay mechanism which is not a decoupled provider, and furthermore, has completely failed to identify "receiving a request for said instrumentation data from said instrumentation data source at said decoupled proxy."

In regard to the element "converting said instrumentation data from a format compatible with said managed code environment to a format compatible with said instrumentation data source, and transmitting said converted instrumentation data to said instrumentation data source," the Office Action cites to Morshed, Col. 8, lines 53-60. As those skilled in the art will appreciate, this passage of Morshed describes converted semi-compiled source code into machine specific source code. However, the Office Action's application of this to the present invention is quite confusing. On the one hand, the Office Action's assertion indicates that the managed code environment is the intermediate source code (which doesn't make sense), and that the instrumentation data source is one of the target machines. As those skilled in the art will appreciate, this is all part and parcel of a compiler, i.e., a single environment. This does not identify instrumentation data in a managed code environment **and** an instrumentation data source executing outside of the managed code environment. Perhaps the final product could be executed outside of a debugging environment, but the Office Action and Morshed do not identify the translation and transmission of information from within the managed code environment to an object executing outside of the managed code environment.

For the reasons described above, applicants assert that the 35 U.S.C. § 102(e) rejection of Claim 10 was in error, should be withdrawn, and the claim allowed.

LAW OFFICES OF  
CHRISTENSEN O'CONNOR JOHNSON KINDNESS<sup>PLLC</sup>  
1420 Fifth Avenue  
Suite 2800  
Seattle, Washington 98101  
206.682.8100

### Claims 11-18

Claims 11-18 depend from, or are multiple-dependent claims upon Claim 10. Applicants assert that for this reason, they are allowable over Morshed. Furthermore, these claims recite similar limitations as those described above in regard to Claims 2-9, and, for those additional reasons, they are further distinguishable from the cited reference. Accordingly, applicants request that the 35 U.S.C. § 102(e) rejection of Claims 11-18 be withdrawn, and the claims allowed.

### Claim 14

In addition to depending from Claim 10, Claim 14 recites additional limitations that further distinguish it from the cited reference. In particular, Claim 14 recites the element: "**one or more providers are queried by said decoupled provider in a round-robin fashion** to identify instances satisfying said request." The Office Action cites to Morshed, Col. 35, lines 45-55, as disclosing this element. Applicants disagree.

This passage of Morshed describes how "control is passed to the monitor DLL when an outgoing request or remote procedure call is made." (Morshed, Col. 35, lines 45-46.) Absolutely no mention is made as to querying providers in a round-robin fashion to identify instances satisfying the request.

Applicants assert that Morshed fails to disclose the addition elements recited above. Accordingly, applicants submit that the 35 U.S.C. § 102(e) rejection of Claim 14 is in error, should be withdrawn, and the claim allowed.

### Claim 19

The Office Action rejected Claim 19 as being anticipated by Morshed. More particularly, the Office Action stated that Claim 19 recites limitations similar in scope to Claim 10, and rejected Claim 19 for the same reasons as described above in regard to Claim 10. Applicants assert that Claim 10 is allowable (as discussed above) and submits that Claim 19 is also

allowable over the cited reference, and requests that the 35 U.S.C. § 102(e) rejection be withdrawn, and the claim allowed.

Claim 20

The Office Action rejected Claim 20 for the same reasons as described in regard to Claim 14. Claim 20 recites similar elements to those in Claim 14. Accordingly, for the reasons discussed above, applicants submit that the 35 U.S.C. § 102(e) rejection of Claim 20 was in error, and that the claim should be allowed.

Claim 21

Claim 21 depends from Claim 19. Accordingly, for the same reasons described above in regard to Claim 19, applicants submit that the cited reference fails to disclose each and every element of Claim 21, especially when Claim 21 is read in combination with Claim 19.

As a preliminary matter, applicants point out that according to the preamble of Claim 19, the instrumentation data source executes **outside** of a managed code environment in which the instrumentation data is found. The Office Action cites to Morshed, Col. 17, lines 1-10, as disclosing all elements of this claim. However, this passage refers to a section of a method/process that is carried out to instrument object code. In other words, the Office Action has failed to identify what it considers to be outside of the managed code environment, as according to this passage, all data writes occur from within the debugging system, and as just discussed, there is no internal/external exchange of information.

Additionally, even assuming that Morshed discloses writing data from an external object to a property of a managed code object, which applicants expressly deny, Claim 21 recites the following elements that are not found in Morshed:

"providing a confirmation from said decoupled provider to said instrumentation data source that said property was written."

LAW OFFICES OF  
CHRISTENSEN O'CONNOR JOHNSON KINDNESS<sup>PLLC</sup>  
1420 Fifth Avenue  
Suite 2800  
Seattle, Washington 98101  
206.682.8100

Nothing in the cited passage of Morshed, particularly Col. 17, lines 1-10, disclose providing a confirmation to the instrumentation data source (executing external to the managed code environment). For these reasons, applicants submit that the 35 U.S.C. § 102(e) rejection of Claim 21 was in error, and that the claim should be allowed.

Claim 22

Applicants assert that Morshed fails to disclose the following elements of Claim 22:

"determining whether said request comprises a request to execute a method on an instance of said instrumentation data"

"in response to determining that said request comprises a request to execute a method, identifying said instance of said instrumentation data and executing said method on said instance," and

"providing a confirmation from said decoupled provider to said instrumentation data source that said method was executed."

In regard to "**determining** whether said request comprises a request to execute a method on an instance of said instrumentation data," the Office Action cites to Morshed, Col. 40, lines 60-65. While this passage includes terms such as request and procedure call, this passage utterly fails to identify any **determination** as to whether a request is one to execute a method on an instance of data. This passage merely discusses making procedure calls, such as a "send\_request call for a remote procedure call."

In regard to the element "in response to determining that said request comprises a request to execute a method, identifying said instance of said instrumentation data and executing said method on said instance," applicants first point out that the cited reference for this element, Morshed, Col. 66, lines 23-30, is not at all associated with the previously cited element, and yet the claim clearly recites that identifying said instance of said instrumentation data is in direct response to the previous determination. Thus, it would appear the Office Action is merely

pointing to passages of Morshed with words that are common to the element, but in which the substantive disclosure is not similar.

In any event, the cited passage of Morshed, Col. 66, lines 23-30, is directed to testing whether a binary flag is set, indicating that a breakpoint (a location in which program execution stops when it is encountered during a debugging session) is set. This is not even close to "identifying said instance of said instrumentation data and executing said method on said instance."

In regard to the element "providing a confirmation from said decoupled provider to said instrumentation data source that said method was executed," the Office Action cites to Morshed, Col. 66, lines 25-27, as disclosing this element. It is true that Morshed discloses the use of a binary flag, as the Office Action indicates. However, it is unrealistic to suppose that a flag indicating whether a particular instruction is associated with a breakpoint can somehow be viewed as providing confirmation to a program executing external to a managed code environment that a method was executed.

For the above described reasons, applicants assert that Morshed fails to disclose each and every element of Claim 22. Accordingly, applicants request that the 35 U.S.C. § 102(e) rejection of Claim 21 be withdrawn, and the claim allowed.

#### Claims 23 and 24

Claims 23 and 24 were rejected for the same reasons as Claims 8 and 9. Accordingly, for the same reasons as described above, applicants assert that the 35 U.S.C. § 102(e) rejection of Claims 23 and 24 was in error, and request that the rejections be withdrawn, and the claims allowed.

#### Claims 25-31

Claim 25 was rejected for similar reasons as Claim 10, as reciting similar elements. Furthermore, Claims 26-31 were rejected under the same rationale. As recited above, applicants

LAW OFFICES OF  
CHRISTENSEN O'CONNOR JOHNSON KINDNESS<sup>PLLC</sup>  
1420 Fifth Avenue  
Suite 2800  
Seattle, Washington 98101  
206.682.8100

submit that Morshed fails to disclose each and every element of these claims. Applicants request that the 35 U.S.C. § 102(e) rejections be withdrawn, and the claims allowed.

Claims 32-35

Claims 32-35 were newly added. Independent Claim 32 includes elements, in varying scope, that are found in Claims 1, 10, and 25. Claims 33-35 depend from independent Claim 32.

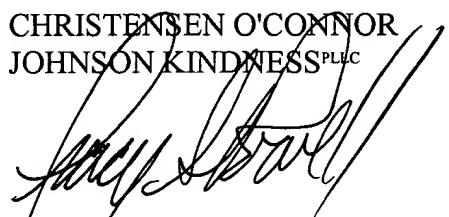
While these claims are not currently rejected, in order to advance prosecution of the application, applicants submit that Morshed fails to disclose each element of these claims, and request their allowance.

CONCLUSION

In view of the amendments and remarks above, applicants respectfully submit that the present application is in condition for allowance. Reconsideration and reexamination of the application, as amended, and allowance of the claims at an early date are solicited. If the Examiner has any questions or comments concerning the foregoing response, the Examiner is invited to contact the applicants' undersigned attorney at the number below.

Respectfully submitted,

CHRISTENSEN O'CONNOR  
JOHNSON KINDNESS<sup>PLLC</sup>



Tracy S. Powell  
Registration No. 53,479  
Direct Dial No. 206.695.1786

I hereby certify that this correspondence is being deposited with the U.S. Postal Service in a sealed envelope as first class mail with postage thereon fully prepaid and addressed to Mail Stop Amendment, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, on the below date.

Date:

March 22, 2005

Lori A. Lewis

TSP:pbe

LAW OFFICES OF  
CHRISTENSEN O'CONNOR JOHNSON KINDNESS<sup>PLLC</sup>  
1420 Fifth Avenue  
Suite 2800  
Seattle, Washington 98101  
206.682.8100